



Source Listings for Computer Code SPIRALI

Incompressible, Turbulent Spiral Grooved Cylindrical and Face Seals

Jed A. Walowitz
Jed A. Walowitz, Inc., Clifton Park, New York

Wilbur Shapiro
Mechanical Technology, Inc., Latham, New York

Restriction changed to Unclassified/Unlimited on July 27, 2005, by authority of the NASA Glenn Research Center, Structures Division.

Export Administration Regulations (EAR) Notice

This document contains information within the purview of the Export Administration Regulations (EAR), 15 CFR 730-774, and is export controlled. It may not be transferred to foreign nationals in the U.S. or abroad without specific approval of a knowledgeable NASA export control official, and/or unless an export license/license exception is obtained/available from the Bureau of Industry and Security, United States Department of Commerce. Violations of these regulations are punishable by fine, imprisonment, or both.

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA Access Help Desk at 301-621-0134
- Telephone the NASA Access Help Desk at 301-621-0390
- Write to:
NASA Access Help Desk
NASA Center for AeroSpace Information
7121 Standard Drive
Hanover, MD 21076



Source Listings for Computer Code SPIRALI

Incompressible, Turbulent Spiral Grooved Cylindrical and Face Seals

Jed A. Walowitz
Jed A. Walowitz, Inc., Clifton Park, New York

Wilbur Shapiro
Mechanical Technology Inc., Latham, New York

Restriction changed to Unclassified/Unlimited on July 27, 2005, by authority of the NASA Glenn Research Center, Structures Division.

Export Administration Regulations (EAR) Notice

This document contains information within the purview of the Export Administration Regulations (EAR), 15 CFR 730-774, and is export controlled. It may not be transferred to foreign nationals in the U.S. or abroad without specific approval of a knowledgeable NASA export control official, and/or unless an export license/license exception is obtained/available from the Bureau of Industry and Security, United States Department of Commerce. Violations of these regulations are punishable by fine, imprisonment, or both.

Prepared under Contract NAS3-25644

National Aeronautics and
Space Administration

Glenn Research Center

Document History

The source and executable files for the CFD Seal Analysis Industrial Codes, of which this is a part, were released as LEW-16582 in 1998. This report was originally published by Mechanical Technology, Inc., in March 1995.

Document Availability Change Notice

NASA/CR—2003-212360

Source Listings for Computer Code SPIRALI Incompressible, Turbulent Spiral Grooved Cylindrical and Face Seals

Jed A. Walowit and Wilbur Shapiro

This document was published in July 2003 with the restriction Export Administration Regulations (EAR). It was changed to Unclassified/Unlimited July 27, 2005, by authority of the NASA Glenn Research Center, Structures Division.

Per the STI Program Office and Code I at Headquarters, you may modify copies in your possession. The restriction notice on the cover, title page, and Report Documentation Page should be boldly crossed out and the above statement printed clearly above or below it.

Trade names or manufacturers' names are used in this report for identification only. This usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

This work was sponsored by the Low Emissions Alternative Power Project of the Vehicle Systems Program at the NASA Glenn Research Center.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100

Available electronically at <http://gltrs.grc.nasa.gov>

```

***** C MAIN PROGRAM FOR COMPUTER CODE SPIRALJ WHICH IS EXTENDS SPIRALJ TO
***** INCLUDE LOCAL PRESSURE JUMPS AT GROOVE-RIDGE DISCONTINUITIES
***** FL /GT1024, SP.RALI.FOR
***** PARAMETER (NDZ=201, NDREG=21)
***** IMPLICIT DOUBLE PRECISION (A-H, O-Z)
***** CHARACTER TITLE*64, PNAME*60, NAME*(3)*60
***** DIMENSION NRSUB(NDREG), ELR(NDREG, RH(2,2))
***** DIMENSION LPL(NDREG), BET(NDREG), DELT(NDREG), DELTI(NDREG),
***** DIMENSION NSG(NDREG), ENGP(NDREG), ZETG(NDREG),
***** DIMENSION RG(NDZ, NDREG), U(NDZ, NDREG), V(NDZ, NDREG),
***** DIMENSION TAU(NDZ, NDREG), CK(4,4), CB(4,4), AR(4,4),
***** DIMENSION FSHP/HAP, HBRL,
***** C THIS COMMON BLOCK PASSES DATA TO USER DEFINABLE FUNCTION FILMSHP
***** COMMON/BFSHP/HAP,HBRL
***** C THIS COMMON BLOCK PASSES TURBULENCE COEFFICIENTS TO
***** C USER DEFINABLE FUNCTIONS FA AND FB
***** COMMON/BFAFB/EMA, EMB, ENB
***** DATA RH20/9.357260*5, 1.03/
***** C INITIALIZE NAMELIST VARIABLES
***** DATA TITLE, IFACE, ISUN, IGROT, NOI, IFLOW// , 5*0/, ,
***** +RO, C, EL, RPM, RPMD, 6*0, D0/,
***** +PLEG, PRIG, VISC, DENS/4*0, D0/
***** +FZD, HOM, NITH, TOLH/0, 10, 1, D-/-
***** +NTV, TOL DUT/30, 1, D-5, 1, D-6/
***** +NREG, NRSUB/1) ELFR(1)/1, 20, 1, D0/
***** +ALP1, BET1, DEL1, ZET1/NDREG*0, D0, NDREG*0, D0, NDREG*0, D0, /
***** +NSG, ZETG/NDREG*0, NDREG*0, D0, /
***** NAMELIST/INPUTS/TITLE, IFACE, ISUN, IGROT, NOI, IFLOW,
***** +RO, C, EL, RPM, RPMD, PLEG, PRIG, VISC, DENS, EMA, EMB, ENB,
***** +HTAP, HBRL, FZD, 1, HOME, NITH, TOLH,
***** +NTV, TOLV DUT
***** +NREG, NRSUB, ELFR, ZET, ALP1, BET1, DELT, NSG, ZETG
***** C USE DATA STATEMENT BELOW TO HARD CODE DEFAULT FILENAMES. BLANK VALUES USED
***** C HERE CAUSES MICROSOFT COMPILER TO TAKE NAMES FROM COMMAND LINE OR ISSUE
***** C PROMPTS FOR THEIR INPUT AT RUN TIME.
***** C UNIT 1 IS INPUT FILE, 2 OUTPUT FILE, 3 PLOT FILE
***** OPEN(1, FILE=FILENAME(1), STATUS='OLD', ERR=9999)
***** OPEN(2, FILE=FILENAME(2), ERR=9999)
***** OPEN(3, FILE=FILENAME(3), ERR=9999)
***** INQUIRE(3, NAME=PNAME)
***** PI=DO ATAN(1, DO)
***** 1CASE=0
***** C INITIALIZE DATA ON COMMON BLOCKS
***** HTAP=0, D0
***** HBR=0, D0
***** EMA=-0.25D0
***** EMB=EMA
***** ENB=ENA
***** READ(1, INPUTS, END=9999)
***** ENGP=NSG/4, DO/PI
***** C CLEAN UP FLAGS
***** IF(IGROT.NE.-1)IGROT=0
***** IF(IFACE.NE.1)IFACE=0
***** IF(ISUN.NE.1)ISUN=0
***** C SET CASE NUMBER AND WRITE CASE NUMBER AND TITLE TO OUTPUT FILE
***** ICASE=ICASE+1
***** IF(ICASE.GT.1)WRITE(2,*), ,

```

```

***** WRITE(2,19)ICASE, TITLE
***** 19 FORMAT(1X, 'CASE', 13, ' ')
***** , AS4/)
***** C PRINT OUT NAMELIST
***** CALL INLIS(2, TITLE, IFACE, ISUN, IGROT, NOI, IFLOW,
***** +RO, C, EL, RPM, RPMD, PLEG, PRIG, VISC, DENS,
***** +EMA, EMA, EMB, ENB, HBRL,
***** +FZD, HOME, NITH, TOLH,
***** +NTV, TOLV DUT, ZET, ALP1, BET1, DELT, NSG, ZETG)
***** CALL OUTSER('STARTING SOLUTION FOR CASE NUMBER ', ICASE)
***** C CHECK UP FRONT FOR ERRORS
***** TER=0
***** IF(CRO.LE.0.D0.OR.CLE.0.D0.OR.CLE.0.D0.OR.VISC.LE.0.D0
***** +.OR.RPM.LT.-1.D-8)TER=8
***** IF(NREG.GT.NDREG)TER=10
***** ELSUM=0, DO
***** DO 81 K=1, NREG
***** ELSUM=ELSUM+ELFR(K)
***** IF(CNRSUB(K).GE.NDZ)TER=9
***** CONTINUE
***** IF(CABS(ELSUM-1.0D0).GT.1.0D-1)TER=11
***** 81 IFC(ELSUM-1.0D0).GT.1.0D-80
***** IFC(IER.NE.0)GO TO 80
***** C GENERATE R AND Z GRIDS
***** ELT=EL/2, DO/RO
***** CALL RZGRID(IFACE, ELT, NREG, NRSUB, ELFR, RG, ZTG)
***** C CHECK ON HOMING IN ON AXIAL LOAD FOR FACE SEAL
***** ITH=0
***** IF(IHOME.NE.2)CNEW=C
***** IF(IFACE.EQ.1.AND.(IHOME.EQ.1.OR.IHOME.EQ.2)
***** +.AND.(ELT.GT.1D-6))ITH=1
***** C LABEL BELOW IS TOP OF HOMING LOOP, USED WHEN ITH>0
***** 88 C=CNNEW
***** IF(IITH.GT.0)CALL OUTSCR, LOAD ITERATION NO., ITH
***** IER=0
***** C GET REFERENCE GAGE PRESSURE AND CHECK DIRECTION OF POISEUILLE FLOW
***** POPPLED-PRIG
***** IDP=1
***** IF(PO.LT.0.)IDP=-1
***** POABSP(PO)
***** C CALCULATE DIMENSIONLESS PRIMARY FILM THICKNESS H, AND TOTAL NO. PTS, MTOT
***** MTOT=0
***** DO 7 K=1, NREG
***** NRT=NRSUB(K)+1
***** MTOT=MTOT+NRT
***** DELT(K)=DELT(K)/C1
***** ENGP(K)=NSG(K)/4, D0/P
***** DO 7 J=1, NRT
***** C ADD SHAPE AND DIVIDE BY C
***** X=ZTG(J, K)/(2, D0*ELT)
***** IF(IFACE.EQ.1.)X=(ELT-1, D0)/(2, D0*ELT)
***** 7 H(J, K)=1, D0+DELT(K)*ALP1(K)+FILMSHP(X)/C1
***** C USE SMALL NUMBER IN PLACE OF 0 DENSITY
***** DN=MAX((1, D-8*RH20*ISUN+1), DENS)
***** C CONVERT ANGULAR VELOCITIES TO RAD/SEC
***** DRCP=PI/30, D0
***** OMFR=RPMD*DRC
***** OM0=RPMD*DRC
***** C CALCULATE VELOCITY AND REYNOLDS NUMBER FOR LAMINAR FLOW
***** VP=L-C1*C1/(12, D0*VISC*EL)
***** V=VP*P0
***** REL=2, D0*C1*VP*DNS/VISC
***** RE=REL
***** RFR=24, D0
***** C GET CHARACTERISTIC AXIAL REYNOLDS NUMBER FOR TURBULENT FLOW
***** IF(REA.GT.1000.D0)THEN

```

```

CALL RECAL(CREL,REA,FBAR,NITV,DUT,IER)
IF(IER.NE.0)GO TO 99
RFBR=REA*FBAR
ENDIF
C COMPUTE AXIAL AND CIRCUMFERENTIAL REYNOLDS NUMBERS TO SELECT RE AND PO
RE=C1*RO*OM*DNS/VISC
IF(REA.GE.REO)THEN
RE=REA
PIN=1.D0
ELSE
TMP=0
PO=PO*OM/2./VPT
PIN=IMP/PO
RE=REO
ENDIF
C MAKE SOME DECISIONS ON HANDLING AXIAL INERTIA
NO1=0
IF(NO1.EQ.0.AND.(C1/EL+TRE/A*RFBR.LT.-01.OR.NO1.GT.0))NO1=1
IF((FLOR*IDIR.EQ.-1.AND.NO1.EQ.0)THEN
IDIR=-IDIR
PIN=-PIN
ENDIF
REC=RE*2.D0*C1/R0*DENS/DNS
IF(NOT EQ.2.REC=0 DO
C COMPUTE CHARACTERISTIC VELOCITY AND DIMENSIONLESS PARAMETERS
V0=RE*VISC/(2.D0*DNS*C1)
P1=VISC*V0*RO/(4.D0*C1*C1*PO)
OMT=OM*RO/VO
OMD=OM*RO/VO
OMD=OMD*RO/VO
C CALCULATE DIMENSIONLESS LOADING
FZND=FZD/PO/RO**2
C PERFORM SEAL COMPUTATIONS
CALL TSEAL(TOLV,NITV,NO1,IFACE,DIR,IROT,NREG,NRSUB,
+RE,REC,PIN,OM,OMD,OMT,OMT,DUT,ZET,ALPI,BETI,DELT1,ENGP,ZETG,
+RG,ZTG,H,U,V,P,TAU,CK,CB,AM,FLO,TOR,W,IMASS,ITER,IER)
PADD=0.D0
IF(PIN.LT.0.D0)PADD=-PIN
JEND=NRSUB(NREG)+1
W=H*P1*PADD*(RG(JEND,NREG)**2-RG(1,1)**2)
RE=RE/DNS*DENS
RE1=RE*(1,1)*ABS(V(1,1))
RE2=RE*(H(1,1)*JEND,NREG)*ABS(V(JEND,NREG))
REO=RE*(H(1,1)*ABSU(1,1)-RG(1,1)*OMT)
REO2=RE*(H(JEND,NREG)*ABS(U(JEND,NREG)-RG(JEND,NREG))*OMT)
C CHECK TO SEE IF RIGHT INLET BOUNDARY WAS USED
IF(IER.EQ.0.AND.NO1.EQ.0.AND.IFLOW*FLO.LT.0.)IER=7
C IF HOMING ON LOAD FOR FACE SEAL, CHECK FOR CONVERGENCE OR TROUBLE
IF((ITH.GT.0.AND.IER.EQ.0.)THEN
IF(ABS(W-FZND)/FZD.LT.10.H)THEN
CONTINUE
ELSE IF(ITH.EQ.NITH)THEN
IER=5
ELSE IF(CK(1,1).LT.1.D-20)THEN
IER=6
ELSE
CNEW=(1.D0-(FZND-W)/CK(1,1))*C1
IF(CHEY/C1.LT.1.D-8)THEN
IER=6
ELSE
ITH=ITH+1
GO TO 88
ENDIF
ENDIF

```

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

SUBROUTINE EMSG(IER)
CHARACTER*78 MSG(11)
C SENDS ERROR MESSAGES TO STD. OUTPUT.
C CALLED BY MAIN PROGRAM
DATA MSG/
      +'INITIAL VELOCITY COMPUTATION DIVERGED',
      +'PRIMARY FLOW COMPUTATION DIVERGED',
      +'MATRIX INVERSION ERROR ENCOUNTERED IN SECOND ORDER SOLUTION',
      +'SPIRAL GROOVE LOCAL FLOW COMPUTATION DIVERGED',
      +'FACE SEAL AXIAL LOAD ITERATION DIVERGED',
      +'NEGATIVE STIFFNESS OR FILM THICKNESS IN AXIAL LOAD ITERATION',
      +'WRONG INLET BOUNDARY WAS USED WITH TRANVERSE INERTIA INCLUDED',
      +'ILLEGAL LENGTH, CLEARANCE, VISCOSITY, PRESSURE OR SPEED ENCOUNTER
      +ED,
      +'MAXIMUM NUMBER OF ALLOWABLE GRID POINTS EXCEEDED',
      +'MAXIMUM NUMBER OF ALLOWABLE REGIONS EXCEEDED',
      +'SUM OF LENGTH FRACTIONS ARE NOT EQUAL TO 1',
      +/
      WRITE(*,* )MSG(IER)
      RETURN
END

```

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

      5
      04/14/1995 13:06      Filename: SPIRAL1.FOR
      6

```

```

SUBROUTINE OUTSCR(MSG,NUM)
C SENDS STATUS MESSAGES TO THE STANDARD OUTPUT UNIT
C CALLED BY MAIN TSEAL
CHARACTER*(* )MSG,CNUM*6,MSG1*78
WRITE(CNUM,'(16')')NUM
DO 5 I=1,6
      1=1
      IF(CNUM(I:I).GT.' ') GO TO 6
      5 CONTINUE
      6 CONTINUE
      L=LEN(MSG)
      MSG1=MSG
      C CONCATINATE NON 0 NUMBER TO STRING
      IF(NUM.EQ.0)THEN
          LT=L
          ELSE
          LT=LT+8-1
          MSG1(L+1:LT)=//CNUM(1:6)
      ENDIF
      WRITE(*,* )MSG1(1:LT)
      RETURN
END

```

```

04/14/1995 13:06 Filename: SPITALLI.FOR Page 8
+E12.4,A7/)
WRITE(1,FILE,64)DDMU,PSIS(K),DENS,PSI4(K)
64  FORMAT(' VISCOSEITY = ',1P,E12.4,A12,
+        ' DENSITY = ',E12.4,A14,')
WRITE(1,FILE,60)ITER
IF(ITER.NE.0)RETURN
40  FORMAT(' ERROR CODE = ',I3,', ITERATIONS IN PRIMARY FLOW = ',I3,
        'ITERATIONS IN SECONDARY FLOW = ',I3)
        WRITE(1,FILE,48)QIN*N13S(K),TOR*FCON*DDC,LBC(K),
        +HP*1HP(K),RE1,REO1,REO2
48  FORMAT(' / FLOW = ',1P,E12.4,A13 '/',
        +' TORQUE = ',1P,E12.4,A10,', FILM POWER LOSS = ',E12.4,A8 '/',
        +' AXIAL REYNOLDS NUMBER = ',1P,E12.4,'/',
        +' CIRC. REYNOLDS NUMBER FOR ROTOR AT SEAL ENDS = ',1P,E12.4,'/',
        +E12.4)
        WRITE(1,FILE,50)
50  FORMAT(' /',      DYNAMIC COEFFICIENTS ( FORCE UNIT / DISP. UNIT
+        +)
        ASSIGN 45 TO KF
        WRITE(1,FILE,KF)XIN(K),YIN(K)
45  FORMAT(' / DISP.',   2A12,
+        +,PHI(RAD),    PSI(RAD),   ' FORCE UNIT ')
        ASSIGN 47 TO KF
DO 100 I=NFP
  IF(LAMASS.EQ.1)THEN
    WRITE(1,FILE,KF)STRK(I,1,FACE+1),
    CCK(I,J)*SCON(I)*XCOR(J,J=1,NFP),KUNIT(I,1,FACE+1,K)
  ELSE
    WRITE(1,FILE,KF)STRK(I,1,FACE+1),
    +(AM(I,J)*SCON(I)*XCOR(J,J=1,NFP),KUNIT(I,1,FACE+1,K)
    +ENDIF
  DO 101 I=1,NFP
    WRITE(1,FILE,KF)STRB(I,1,FACE+1),(CB(I,J)*TCON*SCON(I))/XCON(J),
    +(J=1,NFP),BUNT(I,1,FACE+1,K)
  DO 102 I=1,NFP
    IF(LAMASS.EQ.1)THEN
      WRITE(1,FILE,KF)STRA(I,1,FACE+1),
      +(AM(I,J)*SCON(I)*TCON*I*XCON(J),J=1,NFP),AUNIT(I,1,FACE+1,K)
    ELSE
      WRITE(1,FILE,KF)STRK0(I,1,FACE+1),
      +(CK(I,J)*SCON(I)*XCOR(J),J=1,NFP),KUNIT(I,1,FACE+1,K)
    ENDIF
  CONTINUE
100  CONTINUE
  DO 101 I=1,NFP
    WRITE(1,FILE,KF)STRB(I,1,FACE+1),(CB(I,J)*TCON*SCON(I))/XCON(J),
    +(J=1,NFP),BUNT(I,1,FACE+1,K)
  DO 102 I=1,NFP
    IF(LAMASS.EQ.1)THEN
      WRITE(1,FILE,KF)STRA(I,1,FACE+1),
      +(AM(I,J)*SCON(I)*TCON*I*XCON(J),J=1,NFP),AUNIT(I,1,FACE+1,K)
    ELSE
      WRITE(1,FILE,KF)STRK0(I,1,FACE+1),
      +(CK(I,J)*SCON(I)*XCOR(J),J=1,NFP),KUNIT(I,1,FACE+1,K)
    ENDIF
  CONTINUE
102  CONTINUE
47  RETURN
1000 IF(ITER.EQ.0)THEN
    FCON=P0*DDR*DDR
    QCON=V0*DDR*DDR
    TCON=DDR/V0
    HP=TOR*TCON*DDC*TRPM*1.5866629570-5
    IF((K,EQ.2))HP=HP*6600
    SCON(1)=CON
    SCON(2)=FCON*DDR
    SCON(3)=FCON*DDR
    XCON(1)=DC
    XCON(2)=DDC/DDR
    XCON(3)=DDC/DDR
END IF
NFP=3
  WRITE(1,FILE,80)N13TR(N01+1)
860  FORMAT(' / FACE SEAL',A50)
  WRITE(1,FILE,861)2,DD0*(DDR-DD1),2,DD0*DDR,DDC,IN(K)
861  FORMAT(' / ID, OD NOMINAL FILM THICKNESS = ',
+        +E12.4,'/ ',E12.4,'/ ',E12.4,'/ ',A6 '/')
  WRITE(1,FILE,833)DPL,DPDR,PSI(K)

```

Page 9

04/14/1995 13:06 Filename: SPIRALI.FOR

```

863 FORMAT(' INSIDE, OUTSIDE PRESSURE = ',1P,E12.4,',
+E12.4,A7/)
WRITE(1FILE,64)DDMW,PS1$K),DENS,PS14(K)
WRITE(1FILE,40)ITER
IF(ITER.NE.0)RETURN
FORMAT(' FILE 144)WFCON,LB(K)
WRITE(1FILE,144)WFCON,LB(K)
FORMAT(' AXIAL LOAD TO BALANCE FACE SEAL = ',1P,E12.4,A6)
FORMAT(' RADIAL REYNOLDS NUMBER AT ID, QD = ',1P,E12.4,',
+/' CIRC. REYNOLDS NUMBERS FOR ROTOR AT ID, QD = ',1P,E12.4,',
+E12.4)
FORMAT(' FLOW = ',1P,E12.4,A13,',
+' TORQUE = ',1P,E12.4,A10,' , FILM POWER LOSS = ',E12.4,A8,'/
+' REA1,REA2,REO1,REO2
848 +/' RADIAL REYNOLDS NUMBER AT ID, QD = ',1P,E12.4,',
+' CIRC. REYNOLDS NUMBERS FOR ROTOR AT ID, QD = ',1P,E12.4,',
+E12.4)
FORMAT(' FILE 50)
WRITE(1FILE,50)
ASSIGN 145 TO KF
WRITE(1FILE,KF$ZIN(K)
FORMAT(' DISP ',' A12,
+' PHI (RAD) ',' PS1 (RAD) ',' FORCE UNIT ')
ASSIGN 147 TO KF
DO 500 I=1,NFP
1F(1$AMASS.EQ.1)THEN
  WRITE(1FILE,KF$TRK(I,1$FACE+1),
  +(CK(I,J)*SCON(1)/XCON(J),J=1,NFP),KUNIT(I,1$FACE+1,K)
ELSE
  WRITE(1FILE,KF$TRK(I,1$FACE+1),
  +(AM(I,J)*SCON(1)/XCON(J),J=1,NFP),KUNIT(I,1$FACE+1,K)
+
END IF
500 CONTINUE
DO 501 I=1,NFP
501 WRITE(1FILE,KF$TRB(I,1$FACE+1),(CB(I,J)*TCON*$CON(I)/XCON(J),
+I=1,NFP),BUNIT(I,1$FACE+1,K))
DO 502 I=1,NFP
1F(1$AMASS.EQ.1)THEN
  WRITE(1FILE,KF$TRA(I,1$FACE+1),
  +(CK(I,J)*SCON(1)/TCON#*2/XCON(J),J=1,NFP),KUNIT(I,1$FACE+1,K)
ELSE
  WRITE(1FILE,KF$TRA(I,1$FACE+1),
  +(AM(I,J)*SCON(1)/TCON#*2/XCON(J),J=1,NFP),KUNIT(I,1$FACE+1,K)
+
END IF
502 CONTINUE
FORMAT(1X,A7,1P,3E12.4,3X,A10)
RETURN

```

06/14/1995 13:06 Filename: SPIRAL1.FOR Page 10
 SUBROUTINE INLIST(FILE,TITLE,IFACE,ISIUN,IGROT,NOJ,IFLOW,
 +RO,C,EL,RPM,RPM0,RPMD,PLEG,PRIG,VISC,DENS,
 +ENA,EMB,EMB,HTAP,HBRL,
 +F2D,HOME,NITH,TOLH,
 +NITV,TOV,DUT
 +NREG,NRSUB,ELFR,ZET,ALPI,BETI,DELT,NSG,ZETG)
 C THIS ROUTINE PRINTS OUT THE NAMELIST IN A LEGIBLE MANNER
 C CALLED BY MAIN PROGRAM
 C WRITES TO UNIT NO. I FILE
 IMPLICIT DOUBLE PRECISION (A-H,O-Z)
 PARAMETER (ND=20, NDRREG=21)
 CHARACTER FORM*80, TITLE*64, NTC*2
 DIMENSION NRSUB(NDRREG),ELFR(NDRREG),ZET(NDRREG)
 DIMENSION ALPI(NDRREG), BETI(NDRREG),DELT(NDRREG)
 NT=LEN(TITLE)
 DO 80 I=1,NT
 IT=I+1-1
 IF(ITLE(IT:IT).GT.0)GO TO 81
 CONTINUE
 80 WRITE(NTC,'(12)')IT
 WRITE(ITLE,*1&INPUT/
 FORM='(4X,A8,2X,A1,A//NTC//,A1)'
 WRITE(ITLE,FORM)TITLE=' ',TITLE,' /'/' /' /' /' /'
 WRITE(ITLE,1)'IFACE'=' ',IFACE,' ISIUN'=' ',ISIUN
 WRITE(ITLE,1)'IGROT'=' ',IGROT,' NOJ'=' ',NOJ,'IFLOW'=' ',IFLOW
 WRITE(ITLE,2)'RO'/'EL'=' ',RO/'EL'=' ',EL,'/C'=' ',C
 WRITE(ITLE,2)'RPM'=' ',RPM,'RPM0'=' ',RPM0,'RPMD'=' ',RPMD
 WRITE(ITLE,2)'PLEG'=' ',PLEG,' PRIG'=' ',PRIG,'FZD'=' ',FZD
 WRITE(ITLE,2)'VISC'=' ',VISC,' DENS'=' ',DENS
 WRITE(ITLE,2)'ENA'=' ',ENA,' ENA'=' ',ENA
 WRITE(ITLE,2)'EMB'=' ',EMB,' EMB'=' ',EMB
 WRITE(ITLE,2)'HTAP'=' ',HTAP,' HBRL'=' ',HBRL
 WRITE(ITLE,2)'TOLH'=' ',TOLH,' TOLV'=' ',TOLV
 WRITE(ITLE,3)'HOME'=' ',HOME,' NITH'=' ',NITH,'NITV'=' ',NITV
 NREG=MIN(NREG,5)
 WRITE(ITLE,4)'NREG'=' ',NREG,'NRSUB'=' ',(NRSUB(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,6)(NRSUB(I),I=6,NREG)
 WRITE(ITLE,5)'ELFR'=' ',(ELFR(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(ELFR(I),I=6,NREG)
 WRITE(ITLE,5)'ZET'=' ',(ZET(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(ZET(I),I=6,NREG)
 WRITE(ITLE,5)'ALPI'=' ',(ALPI(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(ALPI(I),I=6,NREG)
 WRITE(ITLE,5)'BETI'=' ',(BETI(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(BETI(I),I=6,NREG)
 WRITE(ITLE,5)'DELT'=' ',(DELT(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(DELT(I),I=6,NREG)
 WRITE(ITLE,8)'NSG'=' ',(NSG(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,9)(NSG(I),I=6,NREG)
 WRITE(ITLE,5)'ZETG'=' ',(ZETG(I),I=1,NREG5)
 IF(NREG.GT.5)WRITE(FILE,7)(ZETG(I),I=6,NREG)
 FORMAT('4X,A8,13,T30,A8,13,T35,A8,13')
 1 FORMAT('4X,A8,1P,E12.4,130,A8,E12.4,155,A8,F12.4)
 2 FORMAT('4X,A8,13,T30,A8,14,T35,A8,I4')
 3 FORMAT('4X,A8,1P,5E12.4')
 4 FORMAT('4X,A8,13,T30,A8,5I4')
 5 FORMAT('4X,A8,1P,5E12.4')
 6 FORMAT('37,5I4')
 7 FORMAT('12X,1P,5I4')
 8 FORMAT('4X,A8,1P,5I4')
 9 FORMAT('4X,A8,1P,5I4')
 RETURN

```

SUBROUTINE TSEALITOL(NITV,NOT,I1DIR,I1GROT1,NREG,NRSUB,
+RE,REC,P1R,PIN,OMT,OMOT,DUT,ZET,ALP,BET,DEL1,ENGP,ZETG,
+RG,ZTG,H,U,V,P,TAU,CK,CB,AM,FLO,TOR,W,IAMASS,ITER,IER)
C TURBULENT SEAL COMPUTATION SUBROUTINE
C CALLED BY MAIN
C CALLS VISOVL TORQ FORCE,KBCAL
C FLAG DEFINITIONS:
C NOI = 1 NEGLECT AXIAL CONVECTIVE INERTIAL TERMS
C IFACE = 1 FACE SEAL
C IIFACE = 0 CYLINDRICAL SEAL
C IDIR = 1 AXIAL FLOW IS KNOWN POSITIVE
C -1 AXIAL FLOW IS KNOWN NEGATIVE
C IGROT = 1 GROOVES ROTATE
C 0 GROOVES STATIONARY
C -1 NO GROOVES (SET BY THIS SUB AND PASSED TO SUPPORTING SUBS)
C IAMASS= 1 CK, CB, AND AM ARE STIFFNESS, DAMPING AND MASS AT 0 FREQUENCY
C 0 CK AND AM ARE DAMPING AND STIFFNESS AT DISTURBANCE FREQUENCY
C IAMASS AND IFR (ERROR CODE ) ARE OUTPUT
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (NDZ=201,NDREG=21)
DIMENSION NRSUB(NDREG),RG(NDZ,NDREG),ZET(NDREG),ZTG(NDZ,NDREG)
DIMENSION ENGP(NDREG),ZETG(NDREG)
DIMENSION ALP(NDREG),SBET(NDREG),CBET(NDREG),VBET(NDREG),
+DELT(NDREG),IGROT(NDREG),UHNG(NDZ,NDREG),VHNG(NDZ,NDREG),
DIMENSION H(NDZ,NDREG),U(NDZ,NDREG),V(NDZ,NDREG),P(NDZ,NDREG),
DIMENSION TAUC(NDZ,NDREG),CK(4,4),CB(4,4),AH(4,4),
+TMP(4,4)
IER=0
P1=4.0*D0*ATAN(1.0)
C ADJUST FLAG AND GET SIN AND COS FOR SPIRAL GROOVE REGION
DO 5 K=1,NREG
  IF (ALP(K).LT.1.0-8.0.R0.OR.(1.0-ALP(K)).LT.1.0-8.0.R0.
+ ABS(SBET(K)).LT.1.0-8.0)THEN
    C IGROT=-1 SIGNIFIES NO GROOVES
    IGROT(K)=-1
  ELSE
    IGROT(K)=IGROT1
    SBET(K)=SIN(BET(K)*P1)/180.0D0
    CBET(K)=COS(BET(K)*P1)/180.0D0
  ENDIF
CONTINUE
C HOME IN ON INITIAL VELOCITY AND GET VELOCITY AND PRESSURE DISTRIBUTIONS
VI=ABS(PIN)
VI=MAX(VI,1.0-3)
CALL QUTSCR(/, FIRST ORDER SOLUTION' 0)
CALL VISOL(YOLV,NITV,VI,NO,IFACE,I1DIR,NREG,NRSUB,
+RE,REC,P1R,PIN,OMT,OMOT,DUT,ZET,
+IGROT,ALP,SBET,CBET,DEL1,ENGP,ZETG,UHG,VHG,
+RG,ZTG,H,U,V,TAU,TOR)
IF (IER .NE. 0)RETURN

```

```

C CALCULATE DIMENSIONLESS SHEAR STRESS AND FLOW AND TORQUE PARAMETERS
FLO=RG(1,1)*H(1,1)**(1,1)*2.0D0
CALL TORQ(NO,IFACE,I1DIR,RE,REC,P1R,OMT,NREG,NRSUB,
+IGROT,ALP,SBET,CBET,DEL1,ENGP,UHG,VHG,
+RG,ZTG,H,U,V,TAU,TOR)
C CALCULATE LOAD UNDER FACE SEAL
IF (IFACE.EQ.1)CALL FORCENREG,NRSUB,RG,ZTG,P,W
C CALCULATE O FREQUENCY STIFFNESS, CK, AND DAMPING
CALL QUTSCR(/, SECOND ORDER SOLUTION' 0)
CALL KBCAL(NO,IFACE,I1DIR,NREG,NRSUB,
+RE,REC,P1R,OMT,DUT,ZET,
+IGROT,ALP,SBET,CBET,DEL1,ENGP,ZETG,UHG,VHG,
+RG,ZTG,H,U,V,CK,CB,IER)
IF (IER .NE. 0)RETURN
IF (ABS(OMT).GT.1.0-8.0)THEN
  C IF NON 0 VALUE OF DISTURBANCE FREQ. AM AND CB WILL CONTAIN STIFFNESS AND

```

04/14/1995 13:06 Filename: SPIRALI.FOR

```

FUNCTION FLMSHP(X)
C THIS IS THE USER DEFINED FILM SHAPE FUNCTION
C X IS THE DISTANCE FROM THE CENTER OF THE SEAL DIVIDED BY THE SEALING
C LENGTH L . - .5 <= X <= .5
C S/(2*L/D) (S IS ZTG IN CODE)
C FOR A SHAFT SEAL X = S/(2*L/D)
C FOR A FACE SEAL X=(S+L/D-1)/(2*L/D)
C FLMSHP IS THE SHAPE OF THE FILM (0 DIMENSIONAL)
C IMPPLICIT DOUBLE PRECISION (A-H,O-Z)
COMMON/BFSHP/HTAP HBRL
FLMSHP=HTAP*X+HBRL*(1.D0-(2.D0*X)**2)
RETURN
END

```

Page 13

04/14/1995 13:06 Filename: SPIRALI.FOR

```

FUNCTION FA(Re,H)
C USER DEFINABLE FRICTION FACTOR FOR MOVING SURFACE
C CALLED BY RECAL TORQ PHISI
C COMMON BLOCK PASSED FROM MAIN PROGRAM
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
COMMON/BFAFRE/ENA,EMB,ENB
FA=MAX(24.D0/RE,ENA*RE**ENA)
C H IS NOT USED NOW BUT MAY BE IN FUTURE FOR TREATING ROUGHNESS
H=H
RETURN
END

```

Page 14

```

FUNCTION FB(RE,H)
C USER DEFINABLE FRICTION FACTOR FOR STATIONARY SURFACE
C CALLED BY RECAL, TORQ, HIPS1
C COMMON BLOCK PASSED FROM MAIN PROGRAM
COMMON/BIAFB/EMA,ENA,ENB,EMB
FB=MAX(24.0D0,RE,ENB*RE**EMB)
C H IS NOT USED NOW BUT MAY BE IN FUTURE FOR TREATING ROUGHNESS
H1=H
RETURN
END

```

```

FUNCTION DELTP(RE,V,H,HSTEP,ZET)
C USER DEFINABLE FUNCTION FOR COMPUTING LOSS COEFFICIENTS
C PRESSURE CHANGE (DOWNSREAM -UPSTREAM) DUE TO SUDDEN CHANGE IN CROSS SECTION
C HSTEP = STEP HEIGHT (H_UPSTREAM-H_DOWNSTREAM)
C CALLED BY UVCAL
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
IF ABS(HSTEP).LT.1.D-8 THEN
  DELTP=0.D0
  GO TO 99
ELSE IF (HSTEP.LT.0.D0)THEN
  C COMPUTE LOSS COEFFICIENT FOR EXPANSION
  ZET1=(1.D0-H/(H+HSTEP))**2
ELSE
  C USE INPUT LOSS COEFFICIENT FOR CONTRACTION (ZET) OR COMPUTE IT FROM RE
  RE1=RE*H*ABS(V)
  ZET1=ZET
ENDIF
DELTP=-(1.D0+ZET1)
IF HSTEP.LT.-1.D8 DELTP=DELTP+(H/(H+HSTEP))***2
DELTP=DELTP*V**V
99
RETURN
END

```

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

FUNCTION DIRFCN(RE,REC,P1R,ONT,VCON,IFACE,1D,R,H,U,DV)
C CALCULATES PRIMARY FLOW DERIVATIVES FOR TAN VEL.. U (ID=1)
C OR PRESSURE P (ID=2)
C CALLED BY UPICAL
C CALLS PHIPSI
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
V=VCON/R/H
CALL PHIPSI(IFACE,RE,REC,ONT,R,H,U,V,PHI,PSI)
IF(ID.EQ.1)DIRFCN=-(PHI*REC)/V
IF(ID.EQ.2)DIRFCN=-P1R*(PSI+REC*V*DV)
RETURN
END

```

Page 17

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

SUBROUTINE RZGRID(IFACE,ELT,NREG,NRSUB,ELFR,RG,ZTG)
C GENERATES R (RZ) AND Z (ZTG) GRIDS
C TO LOCATE STARTING POINT FROM Z ORIGIN ADD INSIDE RADIUS
C FOR FACE SEAL (IFACE=1), SUBTRACT L/D FOR SHAFT SEAL
C R = Z FOR FACE SEAL AND R=1 FOR SHAFT SEAL
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (NDZ=201,NDREG=21)
DIMENSION NRSUB(NDREG),RG(NDZ,NDREG),ELFR(NDREG)
IF(IFACE.NE.-1)FACE=0
ZTG(1,1)=-ELT
RG(1,1)=1.D0
IF(IFACE.EQ.-1)THEN
  ZTG(1,1)=1.D0-2.D0*ELT
  RG(1,1)=ZTG(1,1)
ENDIF
DO 102 KK=1,NREG
  IF(KK.GT.1)THEN
    ZTG(1,KK)=ZTG(NRS+1,KK-1)
    RG(1,KK)=RG(NRS+1,KK-1)
  ENDIF
  NRS=NRSUB(KK)
  DZF=2.D0*ELFR(KK)*ELT/NRS
  DO 102 JJ=1,NRS
    ZTG(JJ+1,KK)=ZTG(JJ,KK)+DZF
    RG(JJ+1,KK)=1.D0
    IF(IFACE.EQ.1)RG(JJ+1,KK)=ZTG(JJ+1,KK)
  CONTINUE
  RETURN
END

```

102

04/14/1995 13:06 Filename: SPIRAL1.FOR

Page 18

```

SUBROUTINE VISOLVTOVL NITY, VI, NOI, IFACE, IDIR, NREG, NRSUB,
+RE, REC, P1R, PIN, OMT, OMOT, DUT, ZET,
+IGROT, ALP, SBET, CBET, DELT, ENGP, ZETG, UHG, VHG,
+RG, ZTG, H, U, V, P, IER)
C SOLVES FOR INLET VELOCITY VI USING NEWTONS METHOD
C ON INPUT VI IS INITIAL GUESS. IER=2 IF NOT CONVERGED.
C CALLED BY TSEAL
C CALLS UPCAL
IMPLICIT DOUBLE PRECISION (A-H,0-Z)
PARAMETER (NDZ=20, NDREG=21)
DIMENSION NRSUB(NDZ, NDREG), RG(NDZ, NDREG), ZET(NDREG), ZTG(NDZ, NDREG)
DIMENSION H(NDZ, NDREG), U(NDZ, NDREG), V(NDZ, NDREG), P(NDZ, NDREG),
DIMENSION ALP(NDREG), SBET(NDREG), CBET(NDREG), DELT(NDREG),
+IGROT(NDREG), UHG(NDZ, NDREG), VHG(NDZ, NDREG)
DIMENSION ENGP(NDREG), ZETG(NDZ, NDREG)
DO 5 J=1, NITV
ITER=J
CALL UPCAL(VI+DUT, NOI, IFACE, IDIR, NREG, NRSUB,
+RE, REC, P1R, PIN, OMT, OMOT, DUT, ZET,
+IGROT, ALP, SBET, CBET, DELT, ENGP, ZETG, UHG, VHG,
+RG, ZTG, H, U, V, P, F2, IER)
CALL UPCAL(VI, NOI, IFACE, IDIR, NREG, NRSUB,
+RE, REC, P1R, PIN, OMT, OMOT, DUT, ZET,
+IGROT, ALP, SBET, CBET, DELT, ENGP, ZETG, UHG, VHG,
+RG, ZTG, H, U, V, P, F1, IER)
DV=F1*DUT/(F2-F1)
VI=VI-DV
VI1=ABS(VI)+TOLV
IF(ABS(DV)/VI1.LT.TOLV)GO TO 6
CONTINUE
IF(NITV.GT.1)IER=2
6 CALL UPCAL(VI, NOI, IFACE, IDIR, NREG, NRSUB,
+RE, REC, P1R, PIN, OMT, OMOT, DUT, ZET,
+IGROT, ALP, SBET, CBET, DELT, ENGP, ZETG, UHG, VHG,
+RG, ZTG, H, U, V, P, F1, IER)
RETURN
END

```

```

SUBROUTINE UVPO(VI, JFACE, IDIR, NREG, MRSUB,
+RE, REC, PIR, PIN, OM, DUT, CBT, DELT, ENGP, UHG, VHG,
+IGROT, IUNG, ALP, SBET, CBET, DELT, ENGP, ZETG, UHG, VHG,
+RG, ZTG, H, U, V, P, PEINIT, IER)
C GENERATES PRESSURE (P) AND VELOCITIES (U,V) BASED ON INITIAL VALUE OF
C V (VI) WITHOUT INERTIA EFFECTS
C CALLED UVPCAL
C CALLS DIRFCN USOLV (NO GROOVES) OR PHIPSG (FOR SPIRAL GROOVES)
C IMPLICIT DOUBLE PRECISION (A-H, O-Z)
PARAMETER (NDZ=20, NDREG=2)
COMMON BLOCK USED LOCALLY IN THIS ROUTINE
COMMON/BRLCL/UHLNDZ, NDREG, VHLNDZ, NDREG)
DIMENSION MRSUB(NDREG), RG(NDZ, NDREG), ZTG(NDZ, NDREG)
DIMENSION H(NDZ, NDREG), U(NDZ, NDREG), V(NDZ, NDREG) P(NDZ, NDREG)
DIMENSION ALP(NDREG), SBET(NDREG), CBET(NDREG), DELT(NDREG),
+IGROT(NDREG), UHG(NDZ, NDREG), VHG(NDZ, NDREG)
DIMENSION ENGP(NDREG), ZETG(NDREG)

C SET UP STARTING CONDITIONS AND LOOPING PARAMETERS FOR PRIMARY FLOW SOLUTION
KST=1
KEN=NREG
IF(IDIR.EQ.-1)THEN
  KST=NREG
  KEN=1
ENDIF
DO 30 K=RST, KEN, IDIR
JST=1
JEN=KST
IF(IDIR.EQ.-1)THEN
  JST=JEN+1
  JEN=2
ENDIF
IF(K.EQ.KST)THEN
  VCON=V1*RG(JST, KST)*IDIR
  V(JST, KST)=V1*IDIR
  P(JST, KST)=PIN
ELSE
  V(JST, K)=VCON/RG(JST, K)/H(JST, K)
  P(JST, K)=P(J1, K1)
ENDIF
IF(IGROT(K).EQ.-1)THEN
  CALL USOLV(RE, QMT, RG(JST, K), H(JST, K), U(JST, K),
V(JST, K), DUT, IUNG, IER)
+    ELSE
+      CALL PHIPSG(1, JFACE, IUNG, RE, REC, QMT, RG(JST, K), H(JST, K),
+      U(JST, K), V(JST, K), UHL(JST, K), VHL(JST, K),
+      ALP(K), SBET(K), CBET(K), DELT(K), ENGP(K), ZETG(K),
+      IGROT(K), PHI, PST, IER)
+    ENDIF
IF(IER.NE.0)RETURN
K1=K
DO 30 J=JST, JEN, IDIR
J1=J+IDIR
RB=.500*(RG(J1, K)+RG(J, K))
HB=.500*(H(J1, K)+H(J, K))
DX=ZTG(J1, K)-ZTG(J, K)
V(J1, K)=VCON/RG(J1, K)/H(J1, K)
IF(IGROT(K).EQ.-1)THEN
  CALL USOLV(RE, QMT, RG(J1, K), H(J1, K), U(J1, K), V(J1, K),
DUT, IUNG, IER)
+    ELSE
+      IF(IER.NE.0)RETURN
      P(J1, K)=(CJ K)+DX*DIFCN( RE, REC, P1R, QMT, VCON, 1FACE, 2,
      RB, HB, .5D0*(U(J1, K)+U(J, K)), 0, D0)
+    ELSE
      VB=VCON/RB/HB
      CALL PHIPSG(1, JFACE, IUNG, RE, REC, QMT, RG(J1, K), H(J1, K),
U(J1, K), V(J1, K), UHL(J1, K), VHL(J1, K),
+    ENDIF
ENDIF

```

```

SUBROUTINE USOLVE(OMT,R,H,U,V,DUT,ILAST,IER)
  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
  C SOLVES FOR EQUILIBRIUM TANGENTIAL VELOCITY WHEN THERE ARE NO GROOVES
  C CALLED BY UPNOI
  C CALLS PHIPSI
  C CALLS PHIPSI1
  IF(ABS(OMT).LT.DUT)THEN
    U=0.0D0
    RETURN
  ENDIF
  DUT=DUT*OMT
  TOL=100.0D0*ABS(DUT)
  IF(ILAST.EQ.0)U=-5.0D0*OMT
  DO 5 I=1,30
    CALL PHIPSI(0,RE,0.0D0,OMT,R,H,U,V,PHI,PSI)
    CALL PHIPSI(0,RE,0.0D0,OMT,R,H,U+DU,V,DU,PSI)
    DLT=PHI+DU/(DPhi-Phi)
    U=U-DLT
    IF(Abs(DLT).LT.TOL)GO TO 6
  5 CONTINUE
  IER=2
  RETURN
END

```

```

SUBROUTINE UPINV1(IFACE,1DIR,NREG,NRSUB,
  +RE,REC,P1R,PIN,OMT,CMT,DUT,ZET,
  +IGROT,UHG,ALP,SET,CBT,DEL,T,ZETG,UHG,VHG,
  +RG,ZTG,H,U,V,P,PEXIT,IER)
  C GENERATES PRESSURE (P) AND VELOCITIES (U,V) BASED ON INITIAL VALUE OF
  C V (V1) WHEN INERTIA IS PRESENT
  C CALLED BY UPVAL
  C CALLS DIRFCN,DELT,PHEPSG (FOR SPIRAL GROOVES)
  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
  PARAMETER (NDZ=201,NREG=21)
  C COMMON BLOCK USED LOCALLY IN THIS ROUTINE
  COMMON/BGRCL/UHL(NDZ,NREG),VHL(NDZ,NREG),
  NRSUB(NDREG),RG(NDZ,NREG)/ZET(NDREG),ZTG(NDZ,NREG)
  DIMENSION H(NDZ,NREG),U(NDZ,NREG),V(NDZ,NREG),P(NDZ,NREG),
  ALP(NDREG),SBET(NDREG),CBET(NDREG),DEL(NDREG),
  +IGROT(NDREG),UHG(NDZ,NREG),VHG(NDZ,NREG),
  DIMENSION ENGP(NDREG),ZETG(NDREG)
  C SET UP STARTING CONDITIONS AND LOOPING PARAMETERS FOR PRIMARY FLOW SOLUTION
  KST=1
  KEN=NREG
  RV0=-5.0D0*REC*P1R
  IF(1DIR.EQ.-1)THEN
    KST=NREG
    KEN=1
  ENDIF
  DO 30 K=KST,KEN,1DIR
  JST=1
  JEN=1
  IF(1DIR.EQ.-1)THEN
    JST=JEN+1
  ENDIF
  JEN=2
  ENDIF
  IF(K.EQ.KST)THEN
    VCON=V1*RG(JST,KST)*H(JST,KST)*1DIR
    V(JST,KST)=V1*1DIR
    UC(JST,KST)=OMT*RG(JST,KST)
    PC(JST,KST)=PIN+
    RV20*DELT(P(1,0),V(JST,KST),H(JST,KST),1.0D10,ZET(KST))
  +
  ELSE
    V(JST,K)=VCON/RG(JST,K)/H(JST,K)
    UC(JST,K)=U(J1,K1)
    P(JST,K)=P(J1,K1)+RV20*DELT(P(1,0),V(JST,K),H(JST,K),ZET(K))
  +
  ENDIF
  K1=K
  DO 30 J=JST,JEN,1DIR
  J1=J+1DIR
  RB=5.0D0*RG(J1,K)*RG(J,K)
  HB=5.0D0*(H(J1,K)+H(J,K))
  DX=ZTG(J1,K)-ZTG(J,K)
  V(J1,K)=VCON/RG(J1,K)/H(J1,K)
  IF(IGROT(K).EQ.-1)THEN
    UI=DIRECN(RE,REC,P1R,OMT,VCON,1FACE,1,
    RB,HB,U(J,K),0.0D0),
    DU1=(DIRFCN(RE,REC,P1R,OMT,VCON,1FACE,1,
    RB,HB,U(J,K)+DU,0.0D0-U1))DU
    U(J1,K)=U(J,K)+DU/(1.0D-5.0D*DX*DU1)
    P(J1,K)=P(J,K)+DU*DIRFCN(RE,REC,P1R,OMT,VCON,1FACE,2,
    RB,HB,.5D0*(U(J1,K)+U(J,K)),(V(J1,K)-V(J,K))/DX)
  +
  ELSE
    VB=VCON/RB/HB
    CALL PHIPSG(1,IFACE,UHG,RE,REC,OMT,RB,HB,
    U(J,K),VB,UHL(J,K),VHL(J,K),ZETG(K),
    ALP(K),SBET(K),CBET(K),DEL(K),ENGP(K),ZETG(K),
    IGROT(K),PHI,PSI,IER)
    IF(IER.NE.0)RETURN
  ENDIF

```

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

U1=-PH1/REC/V(J,K)
CALL PHIPSGO,1,FACE,-1,RE,REC,OMT,RB,HB,
U(J,K)-DUT,VB,UHL(J,K),VHL(J,K),
+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +
ALP(K) SBET(K) CBT(K) DELT(K) ENGP(K) ZETG(K),
IGROT(K) PHI PSI IER
IF(IER.NE.0)RETURN
DUI=(-PH1/REC/VB-U1)/DUT
U(J1,K)=U(J,K)+DX*U1/1.00-500*DX*DU1
CALL PHIPSGO,1,FACE,IUNG,RE,REC,OMT,RB,HB,
.500*(U(J1,K)+U(J,K))VB,URG(J,K),VRG(J,K),
+ + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +
ALP(K) SBET(K) CBT(K) DELT(K) ENGP(K) ZETG(K),
IGROT(K) PHI PSI IER
P(J1,K)=P(J,K)-DI*PIR*(PSI+REC*VB*(V(J1,K)-V(J,K))/DX)
IF(IER.NE.0)RETURN
ENDIF
CONTINUE
PEXIT=P(J1,K1)
RETURN
END
30
```

04/14/1995 13:06 Filename: SPIRAL1.FOR

Page 25

```

SUBROUTINE RECAL(REF,FBAR,NITV,TOLV,IER)
C USES NEWTON ITERATION TO GET CHARACTERISTIC REYNOLDS NUMBER FOR TURBULENT
C POISEUILLE FLOW WITH UNIFORM CLEARANCE
C REL = LAMINAR REYNOLDS NUMBER (INPUT)
C RE = REYNOLDS NUMBER
C IER = ERROR CODE 0 IF OK
C CALLS FA, FB
C CALLED BY MAIN PROGRAM
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
IER=0
REREL
DRE=1.D-6*RE
DO 5 I=1,NITV
11=1
FBAR=(FA(RE,1.00)+FR(RE,1.00))/2.D0
DFBAR=((FA'RE+DR*1.00)+FB*(RE*DR*1.00))/2.D0-FBAR)/DRE
DELT=(RE*RE*FBAR-24.D0*REL)/(RE*RE*DFBAR+2.D0*RE*FBAR)
RE=RE-DELT
IF(ABS(DELT/RE).LT.TOLV)GO TO 6
5 CONTINUE
IER=1
6 RETURN
END
```

Page 26

```

04/14/1995 13:06      Filename: SPIRAL1.FOR
```

```

SUBROUTINE FORCE(NREG,NRSUB,RG,ZTG,P,W)
C COMPUTES DIMENSIONLESS LOAD  $\Psi$ , FROM PRIMARY PRESSURE DISTRIBUTION
C ONLY MEANINGFUL FOR FACE SEAL
C CALLED BY TSEAL
PARAMETER (NDZ=201,NDREG=21)
DIMENSION NRSUB(NDREG),RG(NDZ,NDREG),ZTG(NDZ,NDREG),P(NDZ,NDREG)
W=0.D0
DO 5 K=1, NREG
NR=NRSUB(K)
DO 5 J=1 ,NRS
J=J+1
5 W=+(P(J,K)+P(J1,K))*(RG(J,K)+RG(J1,K))* (ZTG(J1,K))-ZTG(J,K))/4.D0
W=8.D0*ATAN(1.D0)
RETURN
END

```

```

SUBROUTINE TORQ(NOI,1FACE,IDR,RE,REC,P1R,OMT,NREG,NRSUB,
+IGROT,ALP,SBT,CBET,DELT,ENGP,UHG,VHG,
+RG,ZTG,H,U,V,TAU,TOR),
C CALCULATES SHEAR STRESS ON MOVING SURFACE AND TORQUE INTEGRAL
C SHEAR STRESSES AT ARE AT HALF GRID POINTS
C CALLED BY TSEAL
C CALLS FA AND FB
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C PARAMETER (NDZ=201,NDREG=21)
C DIMENSION NRSUB(NDREG),RG(NDZ,NDREG),ZTG(NDZ,NDREG),TAU(NDZ,NDREG)
C DIMENSION H(NDZ,NDREG),ADREG,V(NDZ,NDREG)
C DIMENSION ALP(NDREG),DELT(NDREG),SBET(NDREG),CBET(NDREG),
+IGROT(NDREG),UHG(NDZ,NDREG),VHG(NDZ,NDREG)
C DIMENSION ENGP(NDREG)
C REDUCE COUETTE PART OF SHEAR STRESS BY 3 FOR LAMINAR FLOW
C OR BY 1.2 AS SUGGESTED BY HIRS FOR TURBULENT FLOW
DATA DLAM DHIRS/3.D0,1./200/
VCON=V(1,1)*RG(1,1)*H(1,1)
TOR=0.D0
C NEED TO SET INTEGRATION DIRECTION TO PROPERLY GET UHG AND VHG AT HALF GRID
KST=1
KEN=NREG
IF(IDR.EQ.-1)THEN
  KST=IREG
  KEN=1
ENDIF
DO 4 K=KST,KEN, IDIR
  JST=1
  JEN=NRSUB(K)
  IF(IDR.EQ.-1)THEN
    JST=JEN+1
    JEN=2
  ENDIF
  DO 4 J=JST,JEN, IDIR
    J1=J+1D0
    HBAR=.5D0*(H(J,K)+H(J1,K))
    UBAR=.5D0*(U(J,K)+U(J1,K))
    RBAR=.5D0*(RG(J,K)+RG(J1,K))
    VBAR=VCON/RBAR/HBAR
    DPOR=0.D0
    IF(IGROT(K).EQ.-1)THEN
      RA=REFHBAR*SQR((UBAR-RBAR*OMT)**2+VBAR**2)
      RFA=RA*FA(RA',HBAR)
    ENDIF
    DLM=DHIRS
    I=(ABSRFA-24.D0).LT.1.D-10.DLM=DLM
    TAU(J,K)=P(R*RAFA*(UBAR-RBAR*OMT)/HBAR/DLM
    ELSE
      HR=HBAR-ALP(K)*DELT(K)
      HG=HR*DELT(K)
      UHR=(UBAR*HBAR-UHG(J,K)*ALP(K))/(1.D0-ALP(K))
      VHR=(VBAR*HBAR-VHG(J,K)*ALP(K))/(1.D0-ALP(K))
      RAG=RE*SQR((UHG(J,K)-RBAR*OMT*HR)**2+VHG(J,K)**2)
      RAR=RE*SQR((UHR-RBAR*OMT*HR)**2+VHR**2)
      RAFAG=RA*GA(RAG, HG)
      RAFAR=RA*FA(RAR, HG)
      RBG=RE*SQR(TUHG(J,K)**2+VHG(J,K)**2)
      RBR=RE*SQR(TUHR**2+VHR**2)
      RFBG=RBG*FB(RBG, HG)
      RFBFR=RBR*FB(RBR, HR)
      TAUGA=RA*FA(G*(UHG(J,K)-RBAR*OMT*HG)/HG**2
      TAUGB=RBFG*UHG(J,K)/HG**2
      TAURA=RA*FA*(UHR-RBAR*OMT*HR)/HR**2
      TAURB=RBFR*UHR/HR**2
    C SET COUETTE REDUCTION FACTOR
    DLM=DHIRS
  ENDIF

```

```

IF(ABS(RAFAR-24.D0).LT.1.D-10.OR.ABS(RAFAG-24.D0).LT.1.D-10)
+ DLH=DLAM
C ATTEMPT TO SPLIT OFF COUETTE AND POISEUILLE PORTIONS OF SHEAR STRESS
TAURC=(TAURA+TAURB)/2.D0/DLM
TAUGC=(TAUGA+TAUB)/2.D0/DLM
TAURP=(TAURA+TAURB)/2.D0
TAUR=(TAURA+TAURB)/2.D0
TAUGP=(TAUGA+TAUGB)/2.D0
C IF GROOVES ROTATE CORRECT FOR FORCES AT GROOVE EDGES
IF(IGROT(K).EQ.1)THEN
  TAUGP=TAUGP*(1.D0-2.D0*DELT(K)/HG)
C GET EFFECTS OF LOCAL INERTIA DROP DPCOR
  QN=HBAR*(UBAR-RBAR*OMT)*SBET(K)
  IF(NOT(NE1)QN.FN-HEAR*UBAR*CRET(K)
    DPCOR=REC*ENGPK(K)/RBAR*DELT(PRE,QN/HG,HG,-DELT(K),0.D0)*
  SIGN(1.D0,SBET(K)*QN)
+
  TICOR=(UC(J1,K)-U(J,K))/(ZTG(J1,K)-ZTG(J,K))
  IF(IFACE.EQ.1)TICOR=TICOR+UBAR/RBAR
  TAUGP=TAUGP-DELT(K)*REC*VBARTICOR
ENDIF
TAU(J,K)=PI*R*(ALP(K)*(TAUGC+TAUGP)+*
  (1.D0-ALP(K))*(TAURP+TAURC)+DPCOR)
+
ENDIF
TOR=TOR+TAU(J,K)*RBAR**2*(ZTG(J1,K)-ZTG(J,K))*IDIR
4 CONTINUE
C MULTIPLY BY 2 PI AND CHANGE SIGN SO THAT TORQUE IS + WHEN IT OPPOSES MOTION
TOR=TOR*8.D0*ATAN(1.D0)
RETURN
END

```

```

SUBROUTINE PHIPSGNOI1(FACE,1UNG,RE,REC,OMT,R,H,U,V,UNG,VHG,
+ALP,S,C,DEL,TENG,ZETS,IGROT,PHI,PST,IER),
C GENERATES GLOBAL TURBULENCE FUNCTIONS FOR SPIRAL GROOVES
C CALLED BY UVPNOI_UVPIN,DSOLV
C CALLS PHIPSO,MATINV
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION Q(4),B(4),INDEX(4,3),A(4,4),D(4,4),E(4,4)
DATA A/16*0.D0/
DQ=1.D-6
HRH=ALP*DELT
HG=HR+DELT
UH=**H
VH=**H
IF(1UNG.EQ.0)THEN
  IF(NOT.EQ.1)UH=.5D0*H*R*OMT
  Q(1)=UH
  Q(2)=VH
ELSE
  Q(1)=VH
  Q(2)=UH
ENDIF
Q(3)=(UH-Q(1)*ALP)/(1.D0-ALP)
Q(4)=(VH-Q(2)*ALP)/(1.D0-ALP)
DO 30 L=1,30
  DO 40 I=1,2
    DO 50 J=1,2
      CALL PHIPSC(RE,OMT,R,HG,Q(1),Q(2),B(1),B(2))
      CALL PHIPSC(RE,OMT,R,HG,Q(1)+DQ,Q(2),A(1,1),A(2,1))
      CALL PHIPSC(RE,OMT,R,HG,Q(1),Q(2)+DQ,A(1,2),A(2,2))
      CALL PHIPSC(RE,OMT,R,HG,Q(1),Q(2),B(3),B(4))
      CALL PHIPSC(RE,OMT,R,HR,Q(3)+DQ,Q(4),A(3,1),A(4,1))
      CALL PHIPSC(RE,OMT,R,HR,Q(3)+DQ,Q(4),A(3,2),A(4,2))
      CALL PHIPSC(RE,OMT,R,HR,Q(3)+DQ,Q(4),A(3,3),A(4,3))
      CALL PHIPSC(RE,OMT,R,HR,Q(3)+DQ,Q(4),A(3,4),A(4,4))
      DO 8 K1=1,2
        K2=K1-1
        DO 7 J=1,2
          A(1+K,J+K)-(A(1+K,J+K)-B(1+K))/DQ
          B(1+K)=B(1+K)
        DO 8 J=1,2
          E(I+K)=B(1+K)+(I+K,J+K)*Q(I+K)
          E(1)=C*B(1)+S*B(2)-C*B(3)-S*B(4)
        DO 9 J=1,4
          D(3,J)=ALP*A(1,J)-(1.D0-ALP)*A(3,J)
          D(1,J)=C*A(1,J)+S*A(2,J)-C*A(3,J)-S*A(4,J)
          D(2,1)=S
          D(2,2)=-C
          D(2,3)=-S
          D(2,4)=C
        DO 2 E2=**OMT*DELT*S*IGROT
        IF(NOT.EQ.1)THEN
          E(3)=ALP*B(1)+(1.D0-ALP)*B(3)
        DO 10 J=1,4
          D(3,J)=ALP*A(1,J)-(1.D0-ALP)*A(3,J)
        IF(IFACE.EQ.1)THEN
          COR=REC*VH/R/H**2
          D(3,1)=D(3,1)-ALP*COR
          D(3,3)=D(3,3)-(1.D0-ALP)*COR
        ELSE
          D(3,1)=ALP
          D(3,2)=0.D0
          D(3,3)=1.D0-ALP
          D(3,4)=0.D0
          E(3)=UH
        ENDIF
        D(4,1)=0.D0
        D(4,2)=ALP
        D(4,3)=0.D0
        D(4,4)=1.D0-ALP
      END
    END
  END
END

```

04/14/1995 13:06 Filename: SPIRALI.FOR

```

E(4)=VH
CALL MATINV(D,E,DETER,4,1,1D,4,INDEX)
IF (ID,NE.1) GO TO 99
ICNV=0
EMX=-1.D20
DO 11 I=1,4
  EMX=MAX(EMX,ABS(E(I)-Q(I)))
  Q(I)=E(I)
  IF (EMX.LT.1.D-4) GO TO 31
CONTINUE
GO TO 99
CALL PHISQ(RE,OMT,R,HG,Q(1),Q(2),PHIG,PSIG)
CALL PHISQ(RE,OMT,R,HR,Q(3),Q(4),PHIR,PSIR)
IF (NO(.EQ.1)U-(ALP*Q(1)+(1.D0-ALP)*Q(3))/H
  IF (UHG.NE.-1)THEN
    UHG=Q(1)
    VHG=Q(2)
  ENDIF
ERR=C*PHIG+S*PSIG-C*PHIR-S*PSIR
  IF (ERR.GT.1.D-4)IER=4
  PHI=ALP*PHIG+(1.D0-ALP)*PHIR
  PSI=ALP*PSIG+(1.D0-ALP)*PSIR
C ADD EFFECTS OF LOCAL INERTIA DROP
C NORMAL FLOW
  QN=H*(U-R*OMT*IGR0)*S
  IF (NO(.NE.1)QN=QN*H*V*C
C CONTRACTION LOSS COEFF, ZETG
  PDRP=DELT/RE*QN/HG*HG*0.D0)+DELT/RE, QN/HR, HR, HG-HR, ZETG)
  PHI=PHI*SIGN(.D0, S*QN)*REC*ENG/R*PDRP
  PSI=PSI+SIGN(.D0, C*QN)*REC*ENG/R*PDRP*ABS(C/S)
  IF (FACE,NE.1)RETURN
  IF (NO(.NE.1)PHI=PHI+REC*U*V/R
  PSI=PSI-REC*U/R
  RETURN
  IER=4
END
99

```

Page 31

04/14/1995 13:06 Filename: SPIRALI.FOR

```

SUBROUTINE PHISQ(RE,OMT,R,HG,QS,PHI,PSI)
C GENERATES TURBULENCE FUNCTIONS PHI,PSI BASED ON FLOW RATHER THAN VELOCITY
C EXCLUDES CENTRIFUGAL AND CORIOLIS TERMS FOR FACE SEAL
C CALLS PHISQ
C CALLED BY PHIPSG
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
CALL PHIPSG(RE,0.D0,OMT,R,H,G,T/H,GS/H,PHI,PSI)
RETURN
END

```

```

SUBROUTINE PHIPS1(IFACE,RE,REC,OMT,R,H,U,V,PHI,PSI)
C GENERATES TURBULENCE FUNCTIONS PHI,PSI
C INCLUDES CENTRIFUGAL AND CORIOLIS TERMS FOR FACE SEAL
C CALLS FA, FB
C CALLED BY DIRFCN.DSOLV,PHIPSQ
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
RA=R*H*SQRT((U-R*OMT)**2+V**2)
RB=RE*H*SQRT(U**2+V**2)
CK2=RA*FA(CRA,H)
CK1=CK2*RB*FB(RB,H)
PHI=(CK1**U-CK2*R*OMT)/H**2
PSI=CK1*V/H**2
IF(IFACE.NE.-1)RETURN
PHI=PHIPREC*U*V/R
PSI=PSI-REC*U*U/R
RETURN
END

```

```

SUBROUTINE DSOLV(NO1,IFACE,RE,REC1,P1R,OMT,OMDT,DUT,
+ALP,SBET,CBET,DELT,ENG,P,ZETG,IGROT,Y,
+R1,Z1,H1,U1,V1,UHG,VHG,R2,Z2,H2,U2,V2,FOI,EMI,IER)
C UPDATES Y(I,L,N) TO NEXT Z POSITION
C GENERATES MATRICES FOR DISTURBANCE EQUATIONS AT ONE VALUE OF Z
C DISTURBANCE EQUATIONS ARE IN FORM  $(DY/DZ) = [A1(Y)] + [CB]$ 
C WHERE I=1,2,3 CORRESPONDS TO P,V,U DISTURBANCES
C L=1,2,3 CORRESPONDS TO COMPLEMENTARY SOLUTION, TILT, RADIAL DISTP. WHEN N<3
C N=1,2,3 CORRESPONDS TO EXP(I*(THETA+OMDT*T)),EXP(I*(THETA-OMDT*T)),
C EXP(I*OMDT*T) RESP.
C N=3 IS FOR AXIAL DISTURBANCE APPLIED TO FACE SEAL FOR WHICH CASE
C L=1,2, CORRESPONDS TO COMP. SOL. AND AXIAL DISTP. RESP.
C CALLS PHIPS1,PHIPSG,ESET,CMATIN
C CALLED BY KBCAL
IMPLICIT DOUBLE PRECISION (A-H,O-Z),
COMPLEX*16 A(3,3),B(3,3),T1(3),O1(3),UR,DETER,Y(3,3,3),
+F0I(3,3),EN(3,3),TMP,DF,I
DIMENSION E(3),DE(3),LABEL(3,3)
DATA TI,O1/(0.00,1.00),(0.00,-1.00),(0.00,0.00)/
+REC=REC1
TF(NO1,Eq.1)REC=0.0D0
UB=.5D0*(U-U1)
HB=.5D0*(H-H1)
RB=.5D0*(R-R1)
VB=V*R/H/RB/HB
ZB=.5D0*(Z-Z1)
DZ2=-Z1
DZ2=-.5D0*DZ
DU=(U-U1)/DZ
DV=(V-V1)/DZ
DRH=(RH*(H-H1)+HB*(R-R1))/DZ
REP1R*REC
C CALCULATE TURBULENT FUNCTIONS AND THEIR DERIVATIVES
IF(IGROT.EQ.-1)THEN
CALL PHIPS1(IFACE,RE,REC,OMT,RB,HB,UB,VB,PHI,PSI)
CALL PHIPS1(IFACE,RE,REC,OMT,RB,HB+DUT,UB,VB,PHI,PSI)
CALL PHIPS1(IFACE,RE,REC,OMT,RB,HB,UB-DUT,VB,PHI,PSI)
CALL PHIPS1(IFACE,RE,REC,OMT,RB,HB,UB,DUT,VB,PHI,PSI)
ELSE
UHGB=DHG
VHGB=VHG
CALL PHIPSG(O,IFACE,1,RE,REC,OMT,RB,HB,UB,VB,UHGB,VHGB,
+ALP,SBET,CBET,DELT,ENG,ZETG,IGROT,PHI,PSI,IER)
IF(IER.NE.0)RETURN
CALL PHIPS1(O,IFACE,-1,RE,REC,OMT,RB,HB+DUT,UB,VB,UHGB,VHGB,
+ALP,SBET,CBET,DELT,ENG,ZETG,IGROT,PHI,PSI,IER)
IF(IER.NE.0)RETURN
CALL PHIPS1(O,IFACE,-1,RE,REC,OMT,RB,HB,UB-DUT,VB,UHGB,VHGB,
+ALP,SBET,CBET,DELT,ENG,ZETG,IGROT,PHI,PSI,IER)
CALL PHIPS1(O,IFACE,-1,RE,REC,OMT,RB,HB,UB,DUT,VB,PHI,PSI,IER)
+ALP,SBET,CBET,DELT,ENG,ZETG,GR0T,PHI,PSI,IER)
IF(IER.NE.0)RETURN
ENDIF
PHIH=(PHIH-PHI)/DUT
PSIH=(PSIH-PSI)/DUT
PSIU=(PSIU-PSI)/DUT
PSIV=(PSIV-PSI)/DUT
IF((IFACE.EQ.1.AND.NO1.EQ.1)PSIU=PSIU-REC1/RB**2.D0*UB
NMAX=2+FACE
LMAX=3-1.FACE
DO 5 N=1,NMAX
C SET DISPLACEMENT AMPLITUDES

```

```

CALL ESET(ZB,N,E,DE)
UR=OMD*T1(N)+UB/RB*T1(N)
A(1,1)=(0.0D+0.0D)
A(1,2)=(REC*(DV-VB/RB*DRH)+PSIV*UR*REC1)*P1R
A(1,3)=P1R*PS1U-T1(N)*REP*VB/RB
A(2,1)=(0.0D+0.0D)
A(2,2)=DCMPLX(DRH/RB/RB/HB,0.0D)
A(2,3)=T1(N)/RB
A(3,1)=T1(N)/RB/P1R
A(3,2)=DCMPLX(DU*REC+PHIV,0.0D)
A(3,3)=(PHIU+UR*REC1)

C L=1 IS COMPLEMENTARY SOLUTION
DO 7 J=1,3
    B(1,J)=(0.0D,0.0D)
    DO 6 L=2,LMAX
        TMP=(VB*(RB*DE(L))+IFACE(E(L))+RB*E(L)*(DV+UR))/RB/HB
        B(1,L)=P1R*PS1H*E(L)-REP*VB*TMP
        B(2,L)=TMP
        B(3,L)=DCMPLX(E(L)*PHIV,0.0D)
    CONTINUE

C IF AXIAL (RADIAL) INERTIA IS INCLUDED DIVIDE U EQUATION BY COEFF OF DU/DS
    IF (NOI .NE. 1) THEN
        NEQ=3
        DO 20 J=1,3
            A(3,J)=A(3,J)/REC/VB
        DO 21 L=1,LMAX
            B(3,L)=B(3,L)/REC/VB
        ELSE
            NEQ=2
            DO 23 I=1,2
                A(I,3)=(A(I,3)/A(3,3))
            DO 24 L=1,LMAX
                B(I,L)=B(I,L)-A(I,3)*B(3,L)
            DO 25 J=1,2
                A(I,J)=A(I,J)-A(I,3)*A(3,J)
            ENDIF
        ENDIF
        C REPLACE <B> WITH <B>-[A]<Y>
        DO 22 L=1,LMAX
            DO 22 J=1,NEQ
                B(I,L)=(B(I,L)-A(I,J))*Y(J,L,N)
            C REPLACE [A] WITH [A]+DZ/2*[A]
            DO 8 I=1,NEQ
                DO 8 J=1,NEQ
                    A(I,J)=DZ2*A(I,J)
                    IF(I.EQ.J)A(I,J)=1.0D+A(I,J)
                CONTINUE
            C SOLVE EQUATIONS FOR ALL LMAX RIGHT HAND SIDE VECTORS IN ONE SHOT
            CALL CMATIN(A,B,DETER,NEQ,LMAX,1D_3,LABEL)
            IF(ID .NE. 1)THEN
                IER=3
            RETURN
        ENDIF
    ENDIF
    C CALCULATE NEW <Y>
    DO 9 L=1,LMAX
        DO 10 I=1,NEQ
            Y(I,L,N)=Y(I,L,N)+DZ*B(I,L)
        C UPDATE FORCE AND MOMENT INTEGRALS
        DF1=(C1,L,N)-DZ*B(C1,L)
        FO1(L,N)=FO1(L,N)+DF1
        IF(N.LT.3)EN1(L,N)=EN1(L,N)+2B*DF1
    CONTINUE
    5 RETURN
END

```

```

SUBROUTINE ESET(Z,N,E,DE)
C SETS DISPLACEMENT/TILT AMPLITUDE, E AND SLOPE DE
C L=1,2,3 CORRESPONDS TO COMPLEMENTARY SOLUTION, TILT, RADIAL DISP RESP. WHEN N<3
C L=1,2 CORRESPONDS TO COMP. SOL. AND AXIAL DISP RESP. WHEN N>3
IMPLICIT DOUBLE PRECISION (A-H O-Z)
DIMENSION E(3),DE(3),DEO(3)
DATA EO,DEO/0.0D,1.0D,0.0D/
DO 5 L=1,3
    DE(L)=DEO(L)
    5 E(L)=EO(L)
    IF(N.EQ.3)RETURN
    E(2)=2
    DE(2)=1.0D
    RETURN
END

```

```

SUBROUTINE KBCAL(N01,1FACE,1DIR,NREG,NRSUB,
+RE,REC,P1R,OMT,OMDT1,DUT,ZET,P,ENG,P,ZETG,UHG,VHG,
+IGROT,ALP,SBET,CBET,DELT,ENG,P,ZETG,UHG,VHG,
+RG,ZTG,H,U,V,CK,CB,IER)
C SETS UP BOUNDARY AND CONTINUITY CONDITIONS, SOLVES SECONDARY FLOW PROBLEM
C AND CALCULATES STIFFNESS AND DAMPING COEFFICIENTS.
C V(VI)
C CALLED BY TSEAL
C CALLS DELTP, ESET, AND DSOLV
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (NDZ=201,NDREG=21)
COMPLEX*16 Y(3,3,3),FO1(3,3),EMI(3,3)
DIMENSION ALP(NDREG),SBET(NDREG),CBET(NDREG),DELT(NDREG),
+IGROT(NDREG),UHG(NDZ,NDREG),VHG(NDZ,NDREG),
DIMENSION NSUB(NDREG),RG(NDZ,NDREG),ZET(NDREG),ZTG(NDZ,NDREG),
DIMENSION H(NDZ,NDREG),U(NDZ,NDREG),V(NDZ,NDREG)
DIMENSION ENG(P,NDREG),ZETG(NDREG)
DIMENSION E(3),DE(3),Y(20,5),CK(4,4),CB(4,4)
C INITIAL DISTURBANCES IN INLET VELOCITY (V) FOR COMP AND PARTICULAR SOLUTIONS
DATA Y20/1.D0,0.D0,0.D0/
DATA RV20/-500*REC*P1R
OMDT=OMDT1
C AVOID INDETERMINACY FOR 0 FREQUENCY DISTURBANCE
IF (ABS(OMDT) LT .1.D-4) THEN
  KST=1
  C SET UP STARTING CONDITIONS AND LOOPING PARAMETERS FOR SECONDARY FLOW SOLUTION
  KEN=NREG
  KST=KST+1
  IF (IDIR.EQ.-1) THEN
    KEN=1
  END IF
  NMAX=2+1*FACE
  LMAX=3*FACE
  DO 30 K=IST,KEN,1DIR
    JST=1
    JEN=NRSUB(K)
    IF (IDIR.EQ.-1) THEN
      JST=JEN+1
      JEN=2
    END IF
    HJMP=1.D10
    IF (K.NE.KST) HJMP=H(J1,K1)*H(J2,K)
    C GET DERIVATIVES FOR FLOW LOSS AT JUMP
    IF (NOT NE_1) THEN
      CH1=DELTP(REF,V(J1,K),H(J1,K),H(J2,K)+DUT,HJMP,ZET(K))/DUT
      CH1V=(DELTP(REF,V(J1,K),H(J1,K)+DUT,HJMP,ZET(K))-CH1)/DUT
      CH1V=(DELTP(REF,V(J1,K)+DUT,H(J1,K),HJMP,ZET(K))-CH1)/DUT
    END IF
    DO 5 N=1,NMAX
      C SET DISPLACEMENT AMPLITUDES
      CALL ESE(ZTG,JST,K),N,E,DE)
      DO 5 L=1,LMAX
        IF (K.EQ.KST) THEN
          C SET UP INITIAL OR CONTINUITY CONDITIONS AT START OF EACH REGION
          C (Y) ARE DISTURBANCES IN PRESSURE, AXIAL VELOCITY AND TANGENTIAL VELOCITY
          Y(1,L,N)=0.D0,0.D0
          Y(2,L,N)=DMPLX(Y20(L),0.D0)
          Y(3,L,N)=0.D0,0.D0
          FO1(L,N)=(0.D0,0.D0)
          EM1(L,N)=(0.D0,0.D0)
        ELSE
          Y(2,L,N)=V(JST,K)*H(JST,K)/H(J1,K1)*E(L)
          H(J1,K1)/H(JST,K)*Y(2,L,N)
        + IF (NOT.NE.-1) Y(1,L,N)=Y(1,L,N)+R/20*(-CH1*E(L)+CH1*V(Y(2,L,N))
        + CONTINUE
      END IF
      IF (L.GT.1) THEN
        C STEP THROUGH REGION
        DO J=J+1DIR,JEN,1DIR
          CALL DSOLVN01,1FACE,RE,REC,P,OMT,OMDT,DUT,
          + ALP(K),SBET(K),CBET(K),DELT(K),ENG(P,K),ZETG(K),IGROT(K),
          + RG(J,K),ZTG(J,K),H(J,K),V(J,K),U(J,K),Y(J1,K),
          + Y FO1(E,M,JER)
        IF (IER.NE.0) RETURN
      END IF
    END DO
    CONTINUE
    C COMBINE COMP AND PARTICULAR SOL. TO SATISFY P=0 AT DOWNSTREAM BOUNDARY
    DO 40 L=1,LMAX
      FO1(L,N)=Y(1,L,N)*(C1,N)+FO1(L,N)
      EM1(L,N)=Y(1,L,N)/(C1,N)*EM1(C1,N)+EM1(L,N)
      P12=P/12.D0
      P12=P/12.D0
      IF (1FACE.EQ.1) THEN
        C EXTRACT STIFFNESS AND DAMPING COEFFICIENTS FOR FACE SEAL
        C INITIALIZE STIFFNESS AND DAMPING MATRICES
        DO 41 I=1,4-FACE
          CK(I,J)=0.D0
        DO 41 J=1,4-FACE
          CK(I,J)=P12*DREAL(EM1(2,1)+EM1(2,2))
        END DO
        C AXIAL FORCE DUE TO AXIAL DISPLACEMENT
        CK(1,1)=2.D0*P1*DREAL(FO1(2,5))
        CB(1,1)=2.D0*P1*DIMAG(FO1(2,3))/OMDT
        C MOMENTS DUE TO TILT
        CK(3,3)=P12*DREAL(EM1(2,1)+EM1(2,2))
        CK(2,2)=CK(3,3)
        CB(2,3)=CK(3,3)
        CK(2,2)=P1*DIMAG(EM1(2,1)+EM1(2,2))
        CK(3,2)=CK(2,3)
        CK(2,3)=P12*DREAL(EM1(2,1)-EM1(2,2))/OMDT
        CB(3,2)=-CK(2,3)
        ELSE
          C EXTRACT STIFFNESS AND DAMPING COEFFICIENTS FOR CYLINDRICAL SEAL
          C MOMENTS DUE TO TILT
          CK(4,4)=P12*DREAL(EM1(2,1)+EM1(2,2))
          CK(3,3)=CK(4,4)
          CB(4,4)=P12*DIMAG(EM1(2,1)-EM1(2,2))/OMDT
          CB(3,3)=CK(4,4)
          CK(3,4)=P12*DIMAG(EM1(2,1)+EM1(2,2))
          CK(4,3)=CK(3,4)
          CK(3,4)=-CK(2,3)
          CK(2,3)=-P12*DREAL(EM1(2,1)-EM1(2,2))/OMDT
          CB(4,3)=-CK(3,4)
          CB(3,2)=-CK(4,1)
          CK(4,1)=P12*DREAL(EM1(3,1)+EM1(3,2))
          CK(3,2)=CK(4,1)
          CB(4,1)=P12*DIMAG(EM1(3,1)-EM1(3,2))/OMDT
          CB(3,1)=-P12*DREAL(EM1(3,1)-EM1(3,2))/OMDT
          CB(4,2)=CK(3,1)
          CK(4,2)=P12*DREAL(FO1(2,1)+FO1(2,2))
          CK(2,3)=-CK(1,4)
          CB(1,1)=P12*DIMAG(FO1(2,1)-FO1(2,2))/OMDT
          CB(2,3)=-CK(1,4)
          CK(2,4)=-P12*DIMAG(FO1(2,1)+FO1(2,2))
          CK(1,3)=CK(2,4)
          CB(2,4)=P12*DREAL(FO1(2,1)-FO1(2,2))/OMDT
          CB(1,2)=CK(2,4)
        END IF
      END DO
    END IF
  END DO
  C SET UP BOUNDARY AND CONTINUITY CONDITIONS
  C (Y) ARE DISTURBANCES IN PRESSURE, AXIAL VELOCITY AND TANGENTIAL VELOCITY
  Y(1,L,N)=V(JST,K)*H(JST,K)/H(J1,K1)*E(L)
  EM1(L,N)=(0.D0,0.D0)
  ELSE
    Y(2,L,N)=V(JST,K)*H(JST,K)/H(J1,K1)*Y(2,L,N)
  + IF (NOT.NE.-1) Y(1,L,N)=Y(1,L,N)+R/20*(-CH1*E(L)+CH1*V(Y(2,L,N))
  + CONTINUE
END IF

```

```

C FORCES DUE TO DISPLACEMENT
CK(1,1)=PI2*DREAL(F01(3,1)+F01(3,2))
CK(2,2)=CK(1,1)
CB(1,1)=PI2*DIMAG(F01(3,1)-F01(3,2))/OMDT
CB(2,2)=CB(1,1)
CK(2,1)=PI2*DIMAG(F01(3,1)+F01(3,2))
CK(1,2)=-CK(2,1)
CB(2,1)=PI2*DREAL(F01(3,1)-F01(3,2))/OMDT
CB(1,2)=-CB(2,1)
ENDIF
RETURN

```

```

SUBROUTINE CHATINA(A,B,DETER,N1,M1,LD,N2,INDEX)
C COMPLEX MATRIX INVERTER
C CALLED BY COLPC
IMPLICIT COMPLEX*16(AH,O-Z)
DOUBLE PRECISION AMX
DIMENSION A(N2,N2),B(N2,1),INDEX(N2,3)
EQUIVALENCE (IROW,JROW),(ICOLU,JCOLU), (AMAX, T, SWAP)
M=M1
N=N1
10 DETER =(1.0D0,0.0D0)
DO 20 J=1,N
20 INDEX(J,3) = 0
DO 550 L=1,N
AMAX=0.0D0
DO 105 J=1,N
IF INDEX(J,3)-1) 60, 105, 60
60 DO 100 K=1,N
IF INDEX(K,3)-1) 80, 100, 715
80 IF (AMAX -ABS (A(L,K))) 85, 100, 100
85 IROW=J
ICOLU=K
AMAX =ABS (A(J,K))
100 CONTINUE
105 CONTINUE
110 IF (AMAX)>10, 715, 110
110 INDEX(ICOLU,3) = INDEX(ICOLU,3) +1
INDEX(1,1) IROW
INDEX(1,2)=ICOLU
130 IF ((IROW-1)COLU) 140, 310, 140
140 DETER=-DETER
DO 200 L=1,N
SWAP=A(IROW,L)
A(IROW,L)=A(COLU,L)
200 A(COLU,L)=SWAP
1F(N) 310, 310, 210
210 DO 250 L=1,N
SWAP=B(IROW,L)
B(IROW,L)=B(COLU,L)
250 B(COLU,L)=SWAP
310 PIVOT =A(ICOLU,ICOLU)
1F(PIVOT EQ .0D-0,0.D0)GO TO 715
DETER=DETER+PIVOT
ACOLU,ICOLU)=(1.0D0,0.D0)
DO 350 L=1,N
350 ACOLU,L)=A(ICOLU,L)/PIVOT
1F(N) 380, 380, 360
360 DO 370 L=1,M
370 B(COLU,L)=B(ICOLU,L)/PIVOT
380 DO 550 L=1,N
IF(L-1)COLU) 400, 550, 400
400 T=A(L,1,ICOLU)
ACOLU,ICOLU)=0.0D0,0.0D0)
IF(T EQ .0D-0,0.D0)GO TO 550
430 DO 450 L=1,N
450 ACOL1,L)=ACOL1,L)-A(ICOLU,L)*T
IF(M,550,550, 460
460 DO 500 L=1,M
500 BCOL1,L)=BCOL1,L)-B(ICOLU,L)*T
550 CONTINUE
600 DO 710 I=1,N
L=N+1-I
IF ((INDEX(L,1)-INDEX(L,2)) 630, 710, 630
630 JROW=INDEX(L,1)
JCOLU=INDEX(L,2)
DO 705 K=1,N
SWAP=A(K,JROW)
```

04/14/1995 13:06 Filename: SPIRAL1.FOR

```

A(K,JROW)=A(K,JCOLU)
A(K,JCOLU)=SWAP
705 CONTINUE
710 CONTINUE
ID =1
740 RETURN
715 ID =2
DETER=(0.0D0,0.0D0)
RETURN
END

```

04/14/1995 13:06 Filename: SPIRAL1.FOR

Page 41

```

SUBROUTINE MATINV(A,B,DETER,N1,M1, ID,N2,INDEX)
C REAL MATRIX INVERSION ROUTINE
C CALLED BY HOME COLP
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION A(N2,N2),B(N2,1),INDEX(12,3),
EQUIVALENCE (IROW,JROW), (ICOLU,JCOLU), (AMAX, T, SWAP)
N=M1
N=N1
10 DETER = 1.0D0
DO 20 J=1,N
20 INDEX(J,3) = 0
DO 550 I=1,N
AMAX=0.D0
DO 105 J=1,N
IF (INDEX(J,3)-1) 60, 105, 60
60 DO 100 K=1,N
IF (INDEX(K,3)-1) 80, 100, 715
80 IF (AMAX -ABS (A(J,K))) 85, 100, 100
85 IROW=J
ICOLU=K
AMAX = ABS (A(J,K))
100 CONTINUE
105 CONTINUE
110 INDEX(ICOLU,3) = INDEX(ICOLU,3) +1
INDEX(1,1)=IROW
INDEX(1,2)=ICOLU
130 IF (IROW-ICOLU) 140, 310, 140
140 DETER=DETER
DO 200 L=1,N
200 AC(IRON,L)=(ICOLU,L)
AC(IRON,L)=SWAP
1F(M) 310, 310, 210
210 DO 250 L=1,M
SWAP=B(IRON,L)
250 BC(IRON,L)=B(ICOLU,L)
BC(ICOLU,L)=SWAP
PIVOT =A(ICOLU,ICOLU)
310 IF (PIVOT.EQ.0.D0)GO TO 715
DETER=DETER*PIVOT
AC(ICOLU,ICOLU)=1.D0
DO 350 L=1,N
350 AC(ICOLU,L)=A((ICOLU,L)/PIVOT
1F(M) 380, 380, 360
360 DO 370 L=1,M
370 BC(ICOLU,L)=B((ICOLU,L)/PIVOT
380 DO 550 L=1,N
3F(L-1-ICOLU) 400, 550, 400
400 T=A(L1,ICOLU)
AC(L1,ICOLU)=0.D0
410 IF (T)30, 550, 430
430 DO 450 L=1,N
450 AC(L1,L)=A(L1,L)-A((ICOLU,L)*T
1F(M) 550, 550, 460
460 DO 500 L=1,M
500 BC(L1,L)=B(L1,L)-B((ICOLU,L)*T
550 CONTINUE
600 DO 710 I=1,N
L=N+1
1F (INDEX(L,1)-INDEX(L,2)) 630, 710, 630
630 IROW=INDEX(L,1)
JCOLU=INDEX(L,2)
DO 705 K=1,N
SWAP=A(K,JROW)
AC(K,JROW)=A(K,JCOLU)

```

Page 42

```
A(K,JCOLU)=SNAP  
705 CONTINUE  
710 ID =1  
740 RETURN  
715 ID =2  
DETER=0.D0  
RETURN  
END
```

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.			
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED	
	October 2005	Final Contractor Report	
4. TITLE AND SUBTITLE		5. FUNDING NUMBERS	
Source Listings for Computer Code SPIRALI Incompressible, Turbulent Spiral Grooved Cylindrical and Face Seals		WU-506-42-31-00 WU-590-21-11-00 NAS3-25644	
6. AUTHOR(S)		8. PERFORMING ORGANIZATION REPORT NUMBER	
Jed A. Walowitz and Wilbur Shapiro		E-13616	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
Mechanical Technology, Inc. 968 Albany Shaker Road Latham, New York 12110		NASA CR—2003-212360 MTI 95TM6	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		11. SUPPLEMENTARY NOTES	
National Aeronautics and Space Administration Washington, DC 20546-0001		Jed A. Walowitz, Jed A. Walowitz, Inc., 4 Cypress Point, Clifton Park, New York 12065; and Wilbur Shapiro, Mechanical Technology, Inc., Latham, New York. Project Manager, Anita D. Liang, Aeropropulsion Projects Office, NASA Glenn Research Center, organization code P, 216-977-7439.	
12a. DISTRIBUTION/AVAILABILITY STATEMENT		12b. DISTRIBUTION CODE	
Restriction changed to Unclassified/Unlimited on July 27, 2005, by authority of the NASA Glenn Research Center, Structures Division.		<p>Export Administration Regulations (EAR) Notice This document contains information within the purview of the Export Administration Regulations (EAR), 15 CFR 730-774, and is export controlled. It may not be transferred to foreign nationals in the U.S. or abroad without specific approval of a knowledgeable NASA export control official, and/or unless an export license/license exception is obtained/available from the Bureau of Industry and Security, United States Department of Commerce. Violations of these regulations are punishable by fine, imprisonment, or both.</p> <p>Unclassified-Unlimited Subject Category: 34 Available electronically at http://gltrs.grc.nasa.gov This publication is available from the NASA Center for AeroSpace Information, 301-621-0390.</p>	
13. ABSTRACT (Maximum 200 words)			
<p>This is the source listing of the computer code SPIRALI which predicts the performance characteristics of incompressible cylindrical and face seals with or without the inclusion of spiral grooves. Performance characteristics include load capacity (for face seals), leakage flow, power requirements and dynamic characteristics in the form of stiffness, damping and apparent mass coefficients in 4 degrees of freedom for cylindrical seals and 3 degrees of freedom for face seals. These performance characteristics are computed as functions of seal and groove geometry, load or film thickness, running and disturbance speeds, fluid viscosity, and boundary pressures.</p>			
14. SUBJECT TERMS			15. NUMBER OF PAGES
Face seals; Seals; Cylindrical seals; Leakage; Incompressible; Spiral-grooved; Turbulent; Source listing			28
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

